Rotating Multiple Sets of Labeled Points to Bring Them Into Close Coincidence: A Generalized Wahba Problem

Bisharah Libbus, Gordon Simons, and Yi-Ching Yao

Abstract. While attempting to better understand the 3-dimensional structure of the mammalian nucleus as well as a rigid-body kinematics application, the authors encountered a naturally arising generalized version of the Wahba (1965) problem concerned with bringing multiple sets of labeled points into close coincidence after making appropriate rotations of these sets of labeled points. Our solution to this generalized problem entails the development of a computer algorithm, described and analyzed herein, that generalizes and utilizes an analytic formula, derived by Grace Wahba (1965), for determining space satellite attitudes, that task being to find a suitable rotation that brings one set of m labeled points into close coincidence, in a least-squares sense, with a second set of m labeled points.

1. INTRODUCTION. Given $k+1$ sets $\mathcal{S}_0, \mathcal{S}_1, \dots, \mathcal{S}_k$, each consisting of m n-dimensional labeled points ($k \geq 1, m \geq 2, n \geq 2$), the task is to independently rotate each of the latter k sets so as to bring all of the $k+1$ sets into close coincidence in a least-squares sense. If we denote the points in $\mathcal{S}_j$ by $\{a_{j\ell}, \ell = 1, \dots, m\}$, $j = 0, \dots, k$, then the task is to find k rotation matrices $M_1, \dots, M_k$ that simultaneously minimize the (weighted) loss function

$$S(M_1, \dots, M_k) = \sum_{0 \leq i < j \leq k} \sum_{\ell=1}^{m} w_{ij\ell} ||M_i a_{i\ell} - M_j a_{j\ell}||^2, \tag{1}$$

where $M_0 = I_n$ (the $n \times n$ identity matrix), and where $||v||$ denotes the Euclidean norm of vector v.

When $k = 1$, this is known as "the Wahba problem," thus explaining why we refer to the problem of minimizing (1) as "the generalized Wahba problem" (GWP). Grace Wahba [**17**], as a graduate student, using nothing more than linear algebra and some clever reasoning, obtained an explicit formula for the rotation matrix M_1, while addressing a compelling need by space scientists, in 1965, to estimate satellite attitudes: given two sets of m labeled n-dimensional points $\{a_1, \dots, a_m\}$ and $\{b_1, \dots, b_m\}$, find a rotation matrix M that brings the second set into the best least-squares coincidence with the first, i.e., find a rotation matrix M that minimizes Wahba's (unweighted) loss function $S(M) = \sum_{\ell=1}^{m} ||a_\ell - M b_\ell||^2$. See Figure 1 with $m = 4$ and $n = 3$, where the unit vectors a_ℓ ($\ell = 1, 2, 3, 4$) are representations, in the satellite reference frame, of the directions of four observed objects, and the unit vectors b_ℓ ($\ell = 1, 2, 3, 4$) are representations of the corresponding observations in a known reference frame.

See [**4**] for an elegant analytic solution for the optimizer M in the Wahba problem, the direct use of which can sometimes make an accurate computation of M difficult. Markley [**8**] provides a computationally more accurate approach based on a singular value decomposition of an $n \times n$ matrix with $n = 3$ (in the context of satellite attitude

http://dx.doi.org/10.4169/amer.math.monthly.124.2.149
MSC: Primary 65F30, Secondary 65K05; 15B10

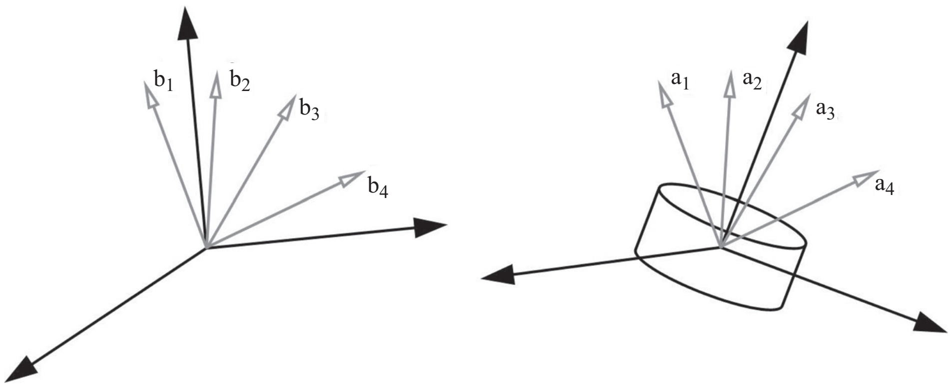

Figure 1. Known (left) and satellite (right) reference frames.

estimation). (For an essentially trivial description of the required computations, see http://en.wikipedia.org/wiki/Wahba%27s_problem.) Note that the optimization problem and related computational methods require no restriction on the vectors a_ℓ and b_ℓ while they are unit vectors in the context of satellite attitude estimation.

Finally, as a segue into applications, we mention that the Wahba problem was extended in the robotics literature (cf. Horn [**5**] and Umeyama [**15**]) to include a translation vector in addition to the rotation matrix, which can be easily reduced to the original Wahba problem. In our formulation of the GWP, we may also include translation vectors along with the rotation matrices, one for each set of labeled points. With minor modifications, our algorithmic approach (to be described and discussed in Section 3) can also be used to deal with this extended version.

The GWP opens up applications to rigid-body kinematics, with "landmarks." For instance, imagine the complicated wrist motion of a baseball pitcher while delivering a curveball to an awaiting batter. The set of landmarks could be chosen in close proximity to the largest carpal (wrist) bone, "capitate." But, in order to secure more accurate calculations, a better choice would appear to be to choose a landmark in close proximity to each of the eight carpal bones, thereby providing a broader base, the only disadvantage being that this collection of landmarks, jointly, is not fully rigid, albeit nearly so. Further, imagine, by some means that we are able to measure, with good precision, the 3-dimensional locations of these eight landmarks at consecutive times $t_0 < t_1 < \cdots < t_k$, for $k \geq 1$. The computational task is to describe, as accurately as possible (in a least-squares sense), the components of linear and rotational motions of the pitcher's wrist over each of the time intervals $[t_0, t_1], [t_1, t_2], \ldots, [t_{k-1}, t_k]$. This task can easily be formulated as an extended version of the GWP. See Spoor and Veldpaus [**14**] and Veldpaus et al. [**16**] for variant versions of the special case $k = 1$ modeling (nearly) rigid-body motion.

A simplified rigid-body example for $k = 2$, $m = 4$, and $n = 3$ appears in Figure 2, with a regular tetrahedron, and with landmarks shown in red, green, yellow and blue (labeled 1, 2, 3, and 4), located at the four vertices. For additional simplification we are suppressing any discussion of linear translations (i.e., we assume that the center of the tetrahedron, shown in black in the figure, is fixed). Thus, there are two rotation matrices to be computed, M_1 and M_2, describing, respectively, the rotations from the first to second and from the first to third "snapshots" (appearing in Figure 2) of the tetrahedron in rotational motion. Now, if the 3-dimensional data describing the locations of all of the vertices were error free, then the minimum value of $S(M_1, M_2)$ would necessarily be equal to 0, and the minimizer would identify precisely accurate rotation matrices

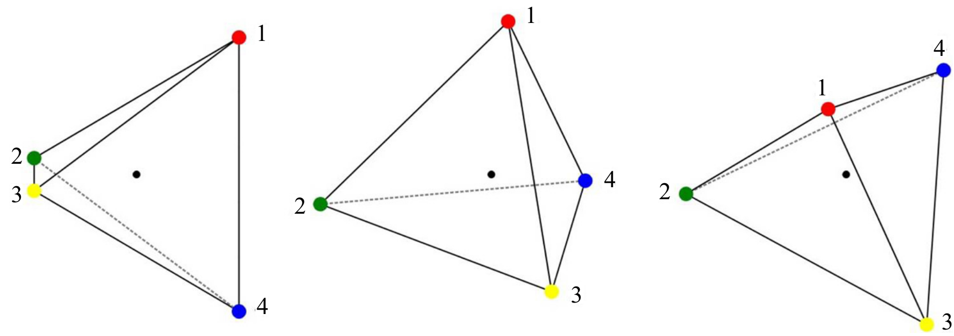

Figure 2. Rotations of a regular tetrahedron.

M_1 and M_2. But measurement errors are to be expected, and the resulting ambiguity in the data can be resolved by minimizing $S(M_1, M_2)$ (that is, by seeking a best least-squares fit of the data, subject to the requirement that M_1 and M_2 are 3×3 rotation matrices).

For rigid-body applications, it makes sense to adjust the weights $w_{ij\ell}$ in (1) appropriately to reflect the fact that one is dealing with a sequence of *contiguous, time-ordered* rotational motions, such as by using larger weights when $j - i$ is small.

Cellular nuclei of eukaryotic organisms, where chromosomes reside, provide the setting for an entirely different application of the GWP. The focus of attention is on the ends of chromosomes, called telomeres. While it is known that telomeres are anchored to the nuclear envelope (cf. Alsheimer et al. [**1**], Crabbe et al. [**2**], and Moens et al. [**9**]), so as to facilitate the required motion of chromosomes within the cellular nucleus during the various phases of cellular activity, the anchoring details are not very well understood. However, specific proteins are being identified as playing key roles in the association of telomeres with the nuclear envelope (cf. Hou et al. [**6**], Kind and van Steensel [**7**], Postberg et al. [**10**], and Schmidt et al. [**12**]). It is tempting to suspect, but presently it cannot be ascertained, whether the arrangement of anchoring points is unique, with each telomere occupying a fixed attachment location to the nuclear envelope *relative to all other telomeres*, an arrangement that is common to all nuclei of the given type. We shall refer to this suspected arrangement as the "fixed anchoring points" (FAP) hypothesis. To be specific, assume that we are observing $k + 1$ cellular nuclei. Now, if it is possible to independently rotate the latter k nuclei, together with their telomere attachments, so as to bring their corresponding telomere locations into close coincidence with the corresponding telomere locations in the first cellular nucleus, this would provide strong evidence in support of the FAP hypothesis. To be even more specific, we might compute a function like $S(M_1, \ldots, M_k)$ in (1), and if the minimum possible value of this function is larger than some specified threshold value, then we might reasonably view this as providing a sound statistical basis for rejecting the FAP hypothesis.

As a practical matter, it is not presently possible to compute the 3-dimensional locations of telomeres within their cellular nuclei. So this appealing approach toward testing the FAP hypothesis is not presently feasible to implement. However, one of the authors of this paper has experimentally secured 2-dimensional projections of the missing 3-dimensional data on telomere locations, and, with this incomplete data, the current authors have been able to convincingly reject the alternative hypothesis that the attachment points of telomeres to the nuclear envelope occur randomly. Unfortunately, this provides scant evidence for the validity of the FAP hypothesis.

Besides the GWP discussed here, we should mention that other kinds of generalizations of the Wahba problem have been discussed in the literature (cf. Shuster [**13**] and Psiaki [**11**]).

Before explaining our successful algorithmic solutions to the GWP, we should point out that attempts by us to directly minimize $S(M_1, \ldots, M_k)$ *analytically*, when $k \geq 2$, have met with failure, a task that appears to be impossible, in general. Something more than analytical reasoning is needed. But, interestingly, Wahba's elegant analytic solution available for $k = 1$ does play a decisive role in the execution of the algorithmic solutions for general k. For the task of minimizing the sum in (1) over all possible rotation matrices $M_1, \ldots, M_k$ can be accomplished algorithmically via repeated suitably designed applications of the analytic methodology used for minimizing $S(M)$ over M (the methodology for the case $k = 1$). The basic idea, in each step, is to hold $k - 1$ of the M's fixed and perform a minimization with respect to the remaining M, doing this repeatedly while cycling through all of the M's. Our experience with the algorithmic approach has been that we routinely obtain a rapid convergence to the desired minimum, obtaining, in the process, very accurate limiting values for the rotation matrices $M_1, \ldots, M_k$. In Section 3 we describe a couple of specific algorithms for cycling through the M's, and establish a convergence result for the second of these. While neither of the two algorithms is guaranteed to solve the minimization problem, again our experience has been that convergence to a global minimum routinely occurs whenever one repeatedly visits all the M's, even if this cycling is performed randomly. In short, we have found that our algorithmic approach is efficacious and robust with respect to cycling variants.

However, as we discuss in Section 3, there are, apparently, rare, exceptional cases of data for which the convergence of $S(M_1, \ldots, M_k)$ can be to something other than the desired global minimum, depending on the starting values of the rotation matrices $M_1, \ldots, M_k$ and the cycling methodology employed. A global minimum can always be found in these cases, but the task is more challenging.

The rest of this paper is organized as follows. As a preliminary to the description and discussion of our algorithmic approach, we present in Section 2 the solution to the Wahba problem via singular value decomposition. In Section 3 we present a couple of algorithms and investigate the convergence issue along with some discussion of numerical results based on extensive simulation studies. Section 4 contains concluding remarks. Proofs of three technical lemmas are relegated to an appendix.

2. SOLUTION TO THE WAHBA PROBLEM. In this section we present the solution to the Wahba problem of minimizing $S(M) = \sum_{\ell=1}^{m} ||a_\ell - Mb_\ell||^2$ over $M \in SO(n)$ via singular value decomposition, where $SO(n)$ denotes the group of all $n \times n$ rotation matrices (orthogonal matrices whose determinants are equal to 1). This is an extension of Markley's [**8**] approach to general n. See also de Ruiter and Forbes [**3**] for discussions of different approaches. To solve the Wahba problem, we need the following lemmas, whose proofs are relegated to the appendix.

Lemma 1. *For a diagonal matrix $D = diag(\delta_1, \ldots, \delta_n)$ with $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_{n-1} \geq |\delta_n|$, we have $\max_{G \in SO(n)} tr(GD) = \sum_{i=1}^{n} \delta_i$, i.e., the maximum value is attained when G is the identity matrix.*

Lemma 2. *For a diagonal matrix $D = diag(\delta_1, \ldots, \delta_n)$ with $\delta_1 \geq \cdots \geq \delta_n \geq 0$, the maximum value of $tr(GD)$ over all orthogonal matrices G with determinant -1 equals $\sum_{i=1}^{n-1} \delta_i - \delta_n$, i.e., the maximum value is attained when $G = J_{(-1)}$, the $n \times n$ identity matrtix with its last diagonal element replaced by -1.*

Since $S(M) = \sum_{\ell=1}^m (||a_\ell||^2 + ||b_\ell||^2) - 2\sum_{\ell=1}^m a_\ell^T M b_\ell$, it is apparent that minimizing $S(M)$ is equivalent to maximizing

$$\widetilde{S}(M) = \sum_{\ell=1}^{m} a_\ell^T M b_\ell = tr(A^T M B), \tag{2}$$

where the notations $(\cdot)^T$ and $tr(\cdot)$ denote a matrix transpose and a matrix trace, respectively, and where $A = (a_1, \ldots, a_m)$ and $B = (b_1, \ldots, b_m)$ are $n \times m$ matrices. Given any $n \times n$ (nonnegative-definite) diagonal matrix $D = diag(\delta_1, \ldots, \delta_n)$ with $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_n \geq 0$, it follows from Lemma 1 that $tr(GD) = \sum_{i=1}^n G_{ii}\delta_i$ is maximized over all rotation matrices $G = (G_{ij}) \in SO(n)$ when $G = J_{(1)} (= I_n)$, the $n \times n$ identity matrix (which attains the maximum value $\sum_{i=1}^n \delta_i$). Moreover, by Lemma 2, $tr(GD)$ is maximized over all orthogonal matrices G whose determinants are equal to -1 when $G = J_{(-1)}$, the $n \times n$ identity matrix with its last diagonal element replaced by -1 (which attains the maximum value $\sum_{i=1}^{n-1} \delta_i - \delta_n$). Now, let UDV^T be a singular value decomposition of the matrix product AB^T, where the diagonal elements of the diagonal matrix D satisfy $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_n \geq 0$ and where U and V are appropriately chosen orthogonal matrices of dimension $n \times n$. Observe, for any rotation matrix M, that

$$\begin{aligned}\widetilde{S}(M) &= tr(A^T M B) = tr((A^T M B)^T) = tr(B^T M^T A) = tr(M^T(AB^T)) \\ &= tr(M^T(UDV^T)) = tr((V^T M^T U)D) = tr(GD),\end{aligned}$$

where $G = V^T M^T U$ is orthogonal with determinant $det(G) = det(U)det(V)$. As M ranges over all rotation matrices, G ranges over all orthogonal matrices with determinant $det(U)det(V)$. It follows that $\widetilde{S}(M)$ is maximized, and $S(M)$ is minimized, over all rotation matrices $M \in SO(n)$ when $M = UJ_{(det(U)det(V))}V^T$. This is the solution to Wahba's (unweighted) problem.

3. ALGORITHMS AND CONVERGENCE RESULTS FOR THE GENERALIZED WAHBA PROBLEM. We now consider the generalized Wahba problem of minimizing $S(M_1, \ldots, M_k)$ in (1) over $(M_1, \ldots, M_k) \in SO(n) \times \cdots \times SO(n)$. Letting $w_{ij\ell} = w_{ji\ell}$ for $i > j$, we may rewrite (1) as

$$\begin{aligned}S(M_1, \ldots, M_k) = &\sum_{0 \leq i < j \leq k} \sum_{\ell=1}^{m} w_{ij\ell}(||a_{i\ell}||^2 + ||a_{j\ell}||^2) \\ &\qquad - \sum_{0 \leq i \neq j \leq k} \sum_{\ell=1}^{m} w_{ij\ell}(M_i a_{i\ell})^T (M_j a_{j\ell}).\end{aligned}$$

Hence, for each fixed $j \in \{1, \ldots, k\}$,

$$\begin{aligned}S(M_1, \ldots, M_k) &= -2 \sum_{i \in \{0,\ldots,k\} \setminus \{j\}} \sum_{\ell=1}^{m} w_{ij\ell}(M_i a_{i\ell})^T (M_j a_{j\ell}) + S_{(-j)} \\ &= -2\, tr(B_j^T M_j A_j) + S_{(-j)},\end{aligned}$$

where $A_j = (a_{j1}, \ldots, a_{jm})$ (matrix of dimension $n \times m$), the ℓth column of the $n \times m$ dimensional matrix B_j is $\sum_{i \in \{0,\ldots,k\} \setminus \{j\}} w_{ij\ell} M_i a_{i\ell}$ ($\ell = 1, \ldots, m$), and $S_{(-j)}$ is a sum of

Table 1. An example for $\mathcal{A}_2$ with $k = 5$:

4	5	2	2	2	2	4	4	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	2	1	4	4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
2	4	4	5	5	5	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	5	1	3	3	3	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
1	5	3	3	1	2	2	2	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4

terms that do not involve M_j. It follows that minimizing $S(M_1, \ldots, M_k)$ over M_j with the other M_i's fixed is equivalent to maximizing $tr(B_j^T M_j A_j)$ over M_j (cf. (2) with A and B replaced by B_j and A_j, respectively), and can be readily solved using any available algorithm (e.g., Markley's singular value decomposition method in Section 2). As described for the fixed index $j \in \{1, \ldots, k\}$, we shall refer to this approach toward reducing the size of $S(M_1, \ldots, M_k)$ as a j-step, applied to a general current state $(M_1, \ldots, M_k)$. Our algorithm for minimizing $S(M_1, \ldots, M_k)$ now takes shape: (i) start with an arbitrary state (configuration) $(M_1^0, \ldots, M_k^0) \in SO(n) \times \cdots \times SO(n)$, called the seed; then (ii) update this state through a sequence of j-steps, updating one rotation matrix at a time, cycling through the possible choices for j in some prescribed manner. What we will call algorithm $\mathcal{A}_1$ uses the trivial cycling strategy: cycling through the indices $\{1, \ldots, k\}$ repeatedly, starting with the index 1. What we will call algorithm $\mathcal{A}_2$ cycles through these indices by choosing at each step a "best possible j-step," i.e., one that reduces the current value of $S(M_1, \ldots, M_k)$ as much as possible. More precisely, if the current (best possible) j-step is for $j = j' \in \{1, \ldots, k\}$, then to determine the next (best possible) j-step, we need to compare all the j-steps with $j \in \{1, \ldots, k\} \setminus \{j'\}$ and choose one that yields the smallest (updated) value of $S(M_1, \ldots, M_k)$. It follows that the two algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ coincide for $k = 2$, while algorithm $\mathcal{A}_2$ is more time consuming for $k > 2$.

For algorithm $\mathcal{A}_2$, unlike for algorithm $\mathcal{A}_1$, the frequency distribution of j-steps performed ($j = 1, \ldots, k$) could become significantly uneven when $k \geq 3$. But empirical evidence indicates otherwise. What we observe is that, after a few j-steps, the pattern of j values chosen by the algorithm starts to repeat according to some permutation of the integers $1, \ldots, k$, continuing in this way until the current values of $S(M_1, \ldots, M_k)$ cease to change (apart from round-off errors). Effectively, convergence has occurred. Table 1 is a typical example for $k = 5$, with the values of j broken up into 40 vertical blocks of 5, describing a total of 200 j-steps. It can be seen that the process of repetition of the permutation (5, 3, 2, 4, 1) begins with the 42nd j-step.

Whatever the cycling strategy used, we shall let $(M_1^r, \ldots, M_k^r)$ denote the state (rotation matrix configuration) at the end of the rth step, $r = 1, 2, \ldots$. Clearly $S(M_1^r, \ldots, M_k^r)$ decreases in r. So two natural questions arise: (1) As $r \to \infty$, does $(M_1^r, \ldots, M_k^r)$ converge? and (2) Does

$$\lim_{r \to \infty} S(M_1^r, \ldots, M_k^r) = \min_{M_1, \ldots, M_k} S(M_1, \ldots, M_k)$$

hold? Since these remain open questions for algorithm $\mathcal{A}_1$, we shall focus our attention on algorithm $\mathcal{A}_2$, addressing them from a theoretical standpoint as well as we presently can.

An illustrative example for the algorithms $\mathcal{A}_i$, for $i = 1, 2$, clarifies what can go wrong. For $k = 2$, $m = 4$, and $n = 3$, we repeatedly computed the minimizing configurations (M_1^*, M_2^*) for randomly generated 3×4 data A_j, for $j = 0, 1, 2$, twice with the same data, starting each trial with a different set of seeds, and checking for possible disagreement. For simplicity we set all of the weights $w_{ij\ell}$ in (1) equal to 1. (Note that

Table 2. An example for $\mathcal{A}_1 = \mathcal{A}_2$ with $k = 2$:

Two Distinct Seed-Dependent Limits for the Same Data

A_0				A_1				A_2			
0.56	0.42	0.99	0.62	−0.09	−0.36	−0.45	−0.90	0.00	0.83	−0.40	−0.40
0.82	0.91	−0.09	0.79	0.72	−0.59	−0.06	−0.36	−0.91	0.11	−0.54	0.44
0.13	0.00	−0.07	−0.01	−0.69	−0.72	0.89	−0.23	0.40	0.55	−0.74	−0.80

M_1 seed			M_2 seed			M_1 limit			M_2 limit		
1.000	0.000	0.000	1.000	0.000	0.000	−0.670	−0.529	0.520	0.049	−0.080	−0.996
0.000	1.000	0.000	0.000	1.000	0.000	−0.427	−0.298	−0.854	0.829	0.560	−0.004
0.000	0.000	1.000	0.000	0.000	1.000	0.607	−0.795	−0.026	0.557	−0.825	0.094
M_1 seed			M_2 seed			M_1 limit			M_2 limit		
0.465	−0.774	−0.429	−0.861	−0.435	0.263	−0.900	0.229	0.370	−0.762	−0.512	−0.396
0.491	0.629	−0.603	0.251	−0.814	−0.525	−0.326	0.209	−0.922	−0.139	−0.468	0.873
0.736	0.070	0.673	0.442	−0.386	0.810	−0.289	−0.951	−0.114	−0.632	0.720	0.285

the two algorithms are the same for $k = 2$.) On the 302nd repetition of this process, we finally encountered a case of disagreement, as shown in Table 2. The pair of seeds used in the second trial are randomly generated rotation matrices. As the table shows, these give rise to a pair of limiting rotation matrices that differ from those resulting from the simple pair of seeds (I_3, I_3) used for the first trial where I_3 is the 3×3 identity matrix. The limiting values of $S(M_1^r, M_2^r)$ (as $r \to \infty$) for this example, are 12.81672 and 12.52939, respectively (with an approximate ratio of 1.02). Extensive empirical studies of this sort, conducted by the authors with randomly generated data and trial seeds, have never produced more than two different limiting configurations (M_1^*, M_2^*) when $k = 2$, $m = 4$, and $n = 3$. So we are confident that the smaller value 12.52939, for this example, corresponds to a genuine global minimum for the sum in (1).

What the two limiting configurations appearing in Table 2 have in common is important to note. They both have the appearance of being a global-minimum configuration in that no further improvement (reduction of $S(M_1^*, M_2^*)$) is possible by the application of an additional j-step. But, of course, one truly is and the other is not a global-minimum configuration. In what follows, we will describe both configurations as "stationary configurations." This is an important concept for us to consider at this point because our methodology naturally leads to the discovery of stationary configurations that might or might not be global-minimum configurations. Whether a stationary configuration is truly a global-minimum configuration depends on the seed and the cycling strategy chosen.

A configuration $(M_1^*, \ldots, M_k^*)$ is said to be *stationary* in $SO(n) \times \cdots \times SO(n)$ (with respect to the sets of points $\{a_{j\ell}\}$ and the weights $w_{ij\ell}$) if for each $j = 1, \ldots, k$,

$$S(M_1^*, \ldots, M_k^*) \leq S(M_1^*, \ldots, M_{j-1}^*, M_j, M_{j+1}^*, \ldots, M_k^*) \quad \text{for all} \quad M_j \neq M_j^*. \quad (3)$$

A configuration $(M_1^*, \ldots, M_k^*)$ is said to be *strictly stationary* if the inequality (3) is strict for each $j = 1, \ldots, k$. Note that $(M_1^*, \ldots, M_k^*)$ is a (strictly) stationary configuration if and only if for each $j = 1, \ldots, k$, M_j^* is the (unique) minimizer of $S(M_1^*, \ldots, M_{j-1}^*, M_j, M_{j+1}^*, \ldots, M_k^*)$ over M_j. Note also that if $(M_1^*, \ldots, M_k^*)$ minimizes $S(M_1, \ldots, M_k)$, then it is a stationary configuration. To address the convergence issues, we need to impose a metric on the (compact) space $SO(n) \times \cdots \times SO(n)$. For convenience, we adopt the norm $|M| := \max_{1 \leq i,j \leq n} |M_{ij}|$ and the metric $d((M_1, \ldots, M_k), (M_1', \ldots, M_k')) := \max\{|M_j - M_j'| : j = 1, \ldots, k\}$.

Theorem 1. *Let $\mathcal{P} = \{P_1, \ldots, P_\nu\}$ be the set of all stationary configurations which is assumed to be finite with cardinality ν. Further assume that for each pair P_i and P_j with $i \neq j$, either $S(P_i) \neq S(P_j)$ or one of P_i and P_j is strictly stationary. Then the sequence of configurations $Q_r := (M_1^r, \ldots, M_k^r)$ generated by algorithm $\mathcal{A}_2$ converges to a stationary configuration (which may depend on the initial configuration $(M_1^0, \ldots, M_k^0)$).*

To prove Theorem 1, we need the following lemma whose proof is relegated to the appendix.

Lemma 3. *Suppose that a subsequence $\{Q_{r_\ell}\}$ converges to $Q' = (M_1', \ldots, M_k')$. Then Q' is stationary and $S(Q') = \lim_{r\to\infty} S(Q_r)$.*

Proof of Theorem 1. To prove that $Q_r = (M_1^r, \ldots, M_k^r)$ converges, it suffices by compactness of $SO(n) \times \cdots \times SO(n)$ to show that any two convergent subsequences $\{Q_{r_\ell}\}$ and $\{Q_{r_u'}\}$ with respective limits $Q' = (M_1', \ldots, M_k')$ and $Q'' = (M_1'', \ldots, M_k'')$ satisfy $Q' = Q''$. By Lemma 3, $Q' = P_i$ and $Q'' = P_j$ for some i and j, and $S(Q') = S(Q'') = c := \lim_{r\to\infty} S(Q_r)$. Suppose $i \neq j$, i.e., $Q' = P_i \neq P_j = Q''$. We will show that this leads to a contradiction. Since $S(P_i) = c = S(P_j)$, we have by the assumption of the theorem that one of P_i and P_j is strictly stationary. Without loss of generality, assume $i = 1$, $j = 2$, and P_1 is strictly stationary. Let $\mathcal{P}' = \{P_h \in \mathcal{P} : S(P_h) = c\}$. Without loss of generality, further assume $\mathcal{P}' = \{P_1, P_2, \ldots, P_{\nu'}\}$ where $2 \leq \nu' \leq \nu$. For each $h = 2, \ldots, \nu'$, P_1 and P_h differ in at least two components since P_1 is strictly stationary and $S(P_1) = S(P_h)$. Write $P_h = (M_1^{(h)}, \ldots, M_k^{(h)})$, for $h = 1, \ldots, \nu'$. (Note that $(M_1', \ldots, M_k') = Q' = P_1 = (M_1^{(1)}, \ldots, M_k^{(1)})$ and $(M_1'', \ldots, M_k'') = Q'' = P_2 = (M_1^{(2)}, \ldots, M_k^{(2)})$.) Let $\varepsilon > 0$ be the smallest one among all *nonzero* values of $|M_i^{(1)} - M_i^{(h)}|$, for $i = 1, \ldots, k$ and $h = 2, \ldots, \nu'$. Consider the neighborhood

$$R_h := \{(O_1, \ldots, O_k) \in SO(n) \times \cdots \times SO(n) : \\ d((O_1, \ldots, O_k), (M_1^{(h)}, \ldots, M_k^{(h)})) < \varepsilon/2\},$$

for $h = 1, \ldots, \nu'$. Clearly $R_1 \cap R_h = \emptyset$, for $h = 2, \ldots, \nu'$. Furthermore, since P_1 and P_h differ in at least two components for $h = 2, \ldots, \nu'$, every configuration in R_1 must differ in two or more components from any configuration in $R_2 \cup \cdots \cup R_{\nu'}$. By Lemma 3, every convergent subsequence of $\{Q_r\}$ converges to some $P_h \in \mathcal{P}'$. It follows that for some $K > 0$,

$$Q_r \in R_1 \cup R_2 \cup \cdots \cup R_{\nu'} \quad \text{for all} \quad r > K \tag{4}$$

(since otherwise there would be a convergent subsequence with a limit $\notin R_1 \cup R_2 \cup \cdots \cup R_{\nu'}$). Since Q_{r_ℓ} converges to $Q' = P_1$, there is an ℓ' such that $r_{\ell'} > K$ and $Q_{r_{\ell'}} \in R_1$. We claim that $(M_1^r, \ldots, M_k^r) \in R_1$ for all $r \geq r_{\ell'} > K$, which contradicts $(M_1^{r_u'}, \ldots, M_k^{r_u'}) \to P_2$. To establish the claim, we proceed by induction. The claim holds for $r = r_{\ell'}$. Suppose $Q_r \in R_1$ for $r \geq r_{\ell'} > K$. We need to show that $Q_{r+1} \in R_1$. Note by (4) that $Q_{r+1} \in R_1 \cup R_2 \cup \cdots \cup R_{\nu'}$. Also by the definition of $\mathcal{A}_2$, Q_r and Q_{r+1} differ only in one component. As noted earlier, every configuration in R_1 must differ in two or more components from any configuration in $R_2 \cup \cdots \cup R_{\nu'}$, implying that $Q_{r+1} \in R_1$. This completes the proof. ■

While for technical reasons we can only address the convergence issue for algorithm $\mathcal{A}_2$ (instead of $\mathcal{A}_1$) as in Theorem 1, this result provides theoretical support of convergence for $\mathcal{A}_1$ as the two algorithms are close cousins. (Recall that $\mathcal{A}_1 = \mathcal{A}_2$ for $k = 2$.) Indeed, we have performed extensive simulation studies with $n = 3$, and for most of the simulation studies we have used algorithm $\mathcal{A}_1$ (instead of $\mathcal{A}_2$), and always observed apparent convergence of $(M_1^r, \ldots, M_k^r)$ as r gets large. It should be remarked that if the sequence of $(M_1^r, \ldots, M_k^r)$ generated by algorithm $\mathcal{A}_1$ converges, then the limiting configuration is necessarily stationary.

Theorem 1 assumes that the number of stationary configurations is finite. To get some idea of how many stationary configurations there can be in the worst possible data situation, we carried out extensive simulation studies for $m = 4, n = 3$, and $k = 2, \ldots, 10$, and found that the maximum numbers of stationary configurations are $2, 2, 3, 3, 4, 4, 4, 4, 5,$ respectively. This is of practical importance when one is worried that the stationary configuration found is not a global-minimum configuration. One can simply use algorithm $\mathcal{A}_1$ repeatedly with randomly generated seeds. For $k = 5$, $m = 4$, and $n = 3$ as an example, one is very likely to end up with the same stationary configuration over and over again, because there is only one stationary configuration (which is necessarily the global-minimum configuration). But if one finds a second stationary configuration, then the configuration that yields the larger value of $S(M_1, \ldots, M_5)$ can be discarded. Continuing, no new stationary configuration is likely to be found, but if a third stationary configuration is encountered, one can again discard the configuration corresponding to the larger value of $S(M_1, \ldots, M_5)$. At this point, one can continue with randomly generated seeds, but this table says that, based on an enormous number of examples we have checked, the maximum number of stationary configurations for $k = 5$ is 3, and new ones will not be found by continuing. So as a practical matter, one is bound by persistence (even in the worst possible data situation) to find the global minimum one seeks. Lest it seem to the reader that this process, as outlined (to make certain that the true global minimum is found), will be time consuming, the actual computational time on a PC will be a few minutes at most, and probably considerably less, simply because algorithm $\mathcal{A}_1$ converges so rapidly. We concede that the above discussion is based solely on empirical evidence without rigorous theoretical justification.

Due to the possible presence of multiple stationary configurations, one can never know for sure if the limiting (stationary) configuration of $(M_1^r, \ldots, M_k^r)$ corresponds to the global minimum of $S(M_1, \ldots, M_k)$. However, it appears to us that this issue is likely to be insignificant *in practice* for the following reasons:

• for the vast majority of the simulated data sets, there appears to be only one stationary configuration which would necessarily correspond to the global minimum of $S(M_1, \ldots, M_k)$;

• in the rare cases when multiple stationary configurations arise, evidence suggests that the $k + 1$ sets of labeled points in the corresponding data set $\{a_{j\ell}, \ell = 1, \ldots, m\}$, for $j = 0, \ldots, k$, cannot be brought into very close coincidence by properly chosen rotation matrices $M_1, \ldots, M_k$, which, of course, is the objective. In view of the rather large size of the global minimum for the example described in Table 2, an inability to obtain a close coincidence of the corresponding labeled points is evident;

• for these exceptional simulated cases of multiple stationary configurations, we have observed that the largest of the $S(M_1^*, \ldots, M_k^*)$ values is only a few percentage points larger than the smallest, namely the one corresponding to the global minimum of $S(M_1, \ldots, M_k)$; cf. the ratio of 1.02 for the example described in Table 2.

4. CONCLUSION. The well-known Wahba problem is to find an optimal rotation that brings one set of labeled points into close coincidence, in a least-squares sense, with a second set of labeled points, which was solved by Wahba [**17**] analytically. Later several effective algorithms were proposed to obtain the solution numerically. In this paper we formulated a generalized version of the Wahba problem which is to find optimal rotations that bring multiple sets of labeled points into close coincidence in a weighted least-squares sense. While there appears to be no analytic solution for this generalized optimization problem, we proposed a couple of algorithms ($\mathcal{A}_1$ and $\mathcal{A}_2$) to solve it numerically. The basic idea, in each step, is to reduce the problem to one that is equivalent to the original Wahba problem and so can be readily solved. While there is no guarantee for the algorithms to find the optimal rotations, we established some convergence results and carried out extensive simulation studies in support of the algorithms. Finally, we note that the Wahba problem was extended in the robotics literature (cf. Horn [**5**] and Umeyama [**15**]) to include a translation vector in addition to the rotation matrix, which can be easily reduced to the original Wahba problem. In our formulation, we may also include translation vectors along with the rotation matrices, one for each set of labeled points. With minor modifications, the algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ can also be used to deal with this extended version.

5. APPENDIX. In the appendix we prove Lemmas 1, 2, and 3.

Proof of Lemma 1. The case $\delta_n \geq 0$ is trivial. We now assume $\delta_n < 0$. Since the function

$$F(\delta_1, \ldots, \delta_n) := \max_{G \in SO(n)} tr(GD)$$

is continuous in $\delta_1, \ldots, \delta_n$, it suffices to prove $F(\delta_1, \ldots, \delta_n) = \sum_{i=1}^n \delta_i$ for $\delta_1 > \cdots > \delta_{n-1} > -\delta_n > 0$. Let $M \in SO(n)$ maximize $tr(GD)$ over $G \in SO(n)$, i.e.,

$$F(\delta_1, \ldots, \delta_n) = tr(MD) = \sum_{i=1}^{n} M_{ii}\delta_i. \tag{5}$$

We claim that $M_{ij} = 0$ for $i \neq j$, from which it follows easily that $M_{ii} = 1$ for all i and $F(\delta_1, \ldots, \delta_n) = \sum_{i=1}^n \delta_i$.

It remains to establish the claim. For each pair (i, j) with $i \neq j$, let $R_{ij}(\theta)$ be the identity matrix with the elements at the four locations (i, i), (i, j), (j, i), and (j, j) replaced by $\cos\theta$, $-\sin\theta$, $\sin\theta$, and $\cos\theta$, respectively. Since $R_{ij}(\theta) \in SO(n)$, we have $MR_{ij}(\theta) \in SO(n)$, and

$$\begin{aligned} f_{ij}(\theta) :&= tr(MR_{ij}(\theta)D) \\ &= \delta_i(M_{ii}\cos\theta + M_{ij}\sin\theta) + \delta_j(M_{jj}\cos\theta - M_{ji}\sin\theta) \\ &\quad + \sum_{h\in\{1,\ldots,n\}\setminus\{i,j\}} \delta_h M_{hh}. \end{aligned}$$

As M maximizes $tr(GD)$ over $G \in SO(n)$, $f_{ij}(\theta)$ attains the maximum value at $\theta = 0$, implying that

$$0 = \frac{d}{d\theta} f_{ij}(\theta)|_{\theta=0} = \delta_i M_{ij} - \delta_j M_{ji}, \quad i.e., \quad \delta_i M_{ij} = \delta_j M_{ji}. \tag{6}$$

By (6) with $i = 1$, we have

$$\delta_1^2 = \sum_{j=1}^{n} \delta_1^2 M_{1j}^2 = \sum_{j=1}^{n} \delta_j^2 M_{j1}^2,$$

which together with $\delta_1^2 > \delta_j^2$ for $j \neq 1$ and $\sum_{j=1}^{n} M_{j1}^2 = 1$ implies that $M_{11}^2 = 1$, which in turn implies that $M_{1j} = M_{j1} = 0$ for $j \neq 1$. Applying (6) repeatedly shows that $M_{ij} = 0$ for $i \neq j$. The proof is complete. ■

Proof of Lemma 2. Note that $tr(GD) = tr(GJ_{(-1)}J_{(-1)}D) = tr(G'D')$, where $G' = GJ_{(-1)}$ and $D' = J_{(-1)}D = diag(\delta_1, \ldots, \delta_{n-1}, -\delta_n)$. As G ranges over all orthogonal matrices with determinant -1, G' ranges over all rotation matrices. The desired result now follows from Lemma 1. ■

Proof of Lemma 3. Since $S(Q_r)$ is decreasing in r, we have

$$c := \lim_{r\to\infty} S(Q_r) = \lim_{\ell\to\infty} S(Q_{r_\ell}) = S(Q').$$

To show that Q' is stationary, suppose to the contrary that for some $1 \leq j \leq k$ and some $M_j^* \in SO(n)$,

$$c^* := S(M_1', \ldots, M_{j-1}', M_j^*, M_{j+1}', \ldots, M_k') < S(M_1', \ldots, M_k') = S(Q') = c.$$

With $\varepsilon = c - c^* > 0$, a standard continuity argument shows that there exists a $\delta > 0$ such that $S(M_1, \ldots, M_{j-1}, M_j^*, M_{j+1}, \ldots, M_k) < c - \varepsilon/2$ whenever $|M_i - M_i'| < \delta$ for all $i \in \{1, \ldots, k\} \setminus \{j\}$. Since $Q_{r_\ell} \to Q'$, there is an ℓ' such that $d(Q_{r_{\ell'}}, Q') < \delta$, i.e., $|M_i^{r_{\ell'}} - M_i'| < \delta$ for all $1 \leq i \leq k$. By the definition of algorithm $\mathcal{A}_2$,

$$\begin{aligned} S(Q_{r_{\ell'}+1}) &\leq \min_{M_j} S(M_1^{r_{\ell'}}, \ldots, M_{j-1}^{r_{\ell'}}, M_j, M_{j+1}^{r_{\ell'}}, \ldots, M_k^{r_{\ell'}}) \\ &\leq S(M_1^{r_{\ell'}}, \ldots, M_{j-1}^{r_{\ell'}}, M_j^*, M_{j+1}^{r_{\ell'}}, \ldots, M_k^{r_{\ell'}}) \\ &< c - \varepsilon/2 < c, \end{aligned}$$

which contradicts the fact that $S(Q_r)$ monotonically decreases to c, completing the proof. ■

REFERENCES

1. M. Alsheimer, E. von Glasenapp, R. Hock, R. Benavente, Architecture of the nuclear periphery of rat pachytene spermatocytes: Distribution of nuclear envelope proteins in relation to synaptonemal complex attachment sites, *Mol. Biol. Cell* **10** (1999) 1235–1245.
2. L. Crabbe, A. J. Cesare, J. M. Kasuboske, J. A. Fitzpatrick, J. Karlseder, Human telomeres are tethered to the nuclear envelope during postmitotic nuclear assembly, *Cell Rep.* **2** (2012) 1521–1529.
3. A. H. J. de Ruiter, J. R. Forbes, On the solution of Wahba's problem $SO(n)$, *J. Astronaut. Sci.* **60** (2013) 1–31.
4. J. L. Farrell, J. C. Stuelpnagel, Problem 65-1: A least squares estimate of spacecraft attitude, *SIAM Rev.* **8** (1966) 384–386.

5. B. K. P. Horn, Closed-form solution of absolute orientation using unit quaternions, *J. Opt. Soc. Amer. A* **4** (1987) 629–642.
6. H. Hou, Z. Zhou, Y. Wang, J. Wang, S. P. Kallgren, T. Kurchuk, E. A. Miller, F. Chang, S. Jia, Csi1 links centromeres to the nuclear envelope for centromere clustering, *J. Cell Biol.* **199** (2012) 735–744.
7. J. Kind, B. van Steensel, Stochastic genome-nuclear lamina interactions, *Nucleus* **5** (2014) 124–130.
8. F. L. Markley, Attitude determination using vector observations and the singular value decomposition, *J. Astronaut. Sci.* **36** (1988) 245–258.
9. P. B. Moens, C. Heyting, A. J. J. Dietrich, W. van Raamsdonk, Q. Chen, Synaptonemal complex antigen location and conservation, *J. Cell Biol.* **105** (1987) 93–103.
10. J. Postberg, S. A. Juranek, S. Feiler, H. Kortwig, F. Jonsson, H. J. Lipps, Association of the telomere-telomere-binding protein complex of hypotrichous ciliates with the nuclear matrix and dissociation during replication, *J. Cell Sci.* **114** (2001) 1861–1866.
11. M. L. Psiaki, Generalized Wahba problems for spinning spacecraft attitude and rate determination, *J. Astronaut. Sci.* **57** (2010) 73–92.
12. J. Schmidt, R. Benavente, D. Hdzic, C. Hoog, C. L. Stewart, M. Alsheimer, Transmembrane protein Sun2 is involved in tethering mammalian meiotic telomeres to the nuclear envelope, *Proc. Natl. Acad. Sci. USA* **104** (2007) 7426–7431.
13. M. D. Shuster, The generalized Wahba problem, *J. Astronaut. Sci.* **54** (2006) 245–259.
14. C. W. Spoor, F. E. Veldpaus, Rigid body motion calculated from spatial co-ordinates of markers, *J. Biomech.* **13** (1980) 391–393.
15. S. Umeyama, Least-squares estimation of transformation parameters between two point patterns, *IEEE Trans. Pattern Anal. Mach. Intell.* **13** (1991) 376–380.
16. F. E. Veldpaus, H. J. Woltring, L. J. M. G. Dortmans, A least-squares algorithm for the equiform transformation from spatial marker co-ordinates, *J. Biomech.* **21** (1988) 45–54.
17. G. Wahba, Problem 65-1: A least squares estimate of spacecraft attitude, *SIAM Rev.* **7** (1965) 409.

BISHARAH LIBBUS is a retired geneticist. He received his Master's degree in biology from the American University of Beirut, Lebanon (AUB), and his Ph.D. in genetics from the University of Missouri, Columbia. He was a postdoc at the Johns Hopkins University before holding various faculty positions at Haigazian College and AUB in Beirut, the University of Vermont, and a senior visiting scientist at the National Institutes of Health. His interest in chromosome order grew out of his study of mammalian meiosis and male chromosome organization.
401 Ironwoods Dr., Chapel Hill, NC 27516
blibbus@gmail.com

GORDON SIMONS is a retired professor of statistics from the University of North Carolina, where he taught statistics and probability for 38 years. He received undergraduate and Master's degrees in mathematics and a Ph.D. in statistics, all from the University of Minnesota, and then was a postdoc at Stanford University for two years before coming to the University of North Carolina in 1968 to teach, and to direct the education and research of graduate students in statistics. He chaired the Department of Statistics for seven years during his tenure at UNC.
Department of Statistics and Operations Research, University of North Carolina, Chapel Hill NC 27599-3260
gsimons@live.unc.edu

YI-CHING YAO received an undergraduate degree in electrical engineering from National Taiwan University in 1976 and a Ph.D. in mathematics (specialized in statistics) from MIT in 1982. He was with the Department of Statistics, Colorado State University during 1983–1995. Since 1995, he has been a research fellow at the Institute of Statistical Science, Academia Sinica, Taiwan. His research interests are in the areas of applied probability and statistics. He served on the editorial board for the journals of *the Annals of Statistics, Bernoulli, Sankhyā*, and *Statistica Sinica*.
Institute of Statistical Science, Academia Sinica, Taipei 115, Taiwan, ROC
yao@stat.sinica.edu.tw